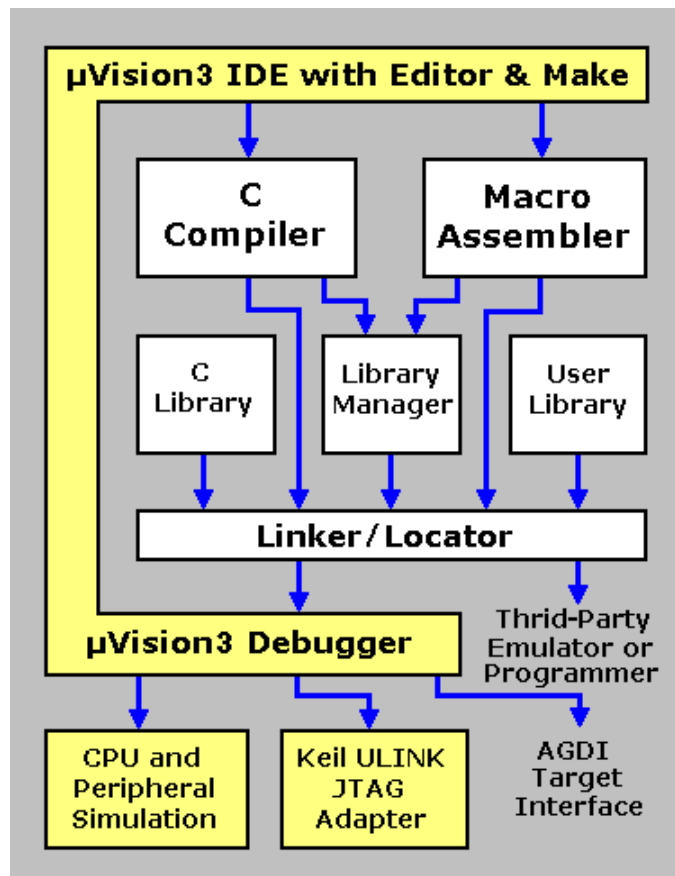


# Notice d'utilisation du logiciel µVision3 de KEIL

## 1- Introduction :

Lors des TP microcontrôleur, nous utilisons l'IDE (Integrated Development Environment) µVision3 de Keil. Ce document résume l'emploi de ce logiciel ainsi que les principes qui régissent son utilisation.



## 2- Du fichier en langage C au programme exécutable

Pour pouvoir générer un exécutable à partir d'un fichier écrit en langage C quatre étapes sont nécessaires :

- Une première étape de compilation qui permet de convertir le fichier écrit en langage C en un fichier assembleur écrit avec les mnémoniques assembleurs du microcontrôleur cible (ici le ARM7 LPC2378). Cette étape est réalisée à l'aide du logiciel armcc.
- Une étape d'assemblage, qui convertit le fichier assembleur en un fichier écrit en binaire, ce fichier est appelé fichier objet. Cette étape est réalisée à l'aide du logiciel armasm.
- Une étape d'édition de lien qui réalise le mappage en mémoire du programme et des données et qui réalise la résolution des références lorsque plusieurs fichiers objets sont utilisés pour générer un exécutable. L'édition de lien génère un nouveau fichier, appelé fichier objet absolu parce qu'il contient les codes binaires et les adresses de mappage de ces codes et des données. Cette étape est réalisée par le logiciel armLink.

- Une étape de conversion du fichier objet absolu en un fichier téléchargeable dans une mémoire de type flash, nous utiliserons le format HEX défini par Intel. Cette étape est réalisée via le logiciel OHA.

### 3- Création d'un nouveau projet

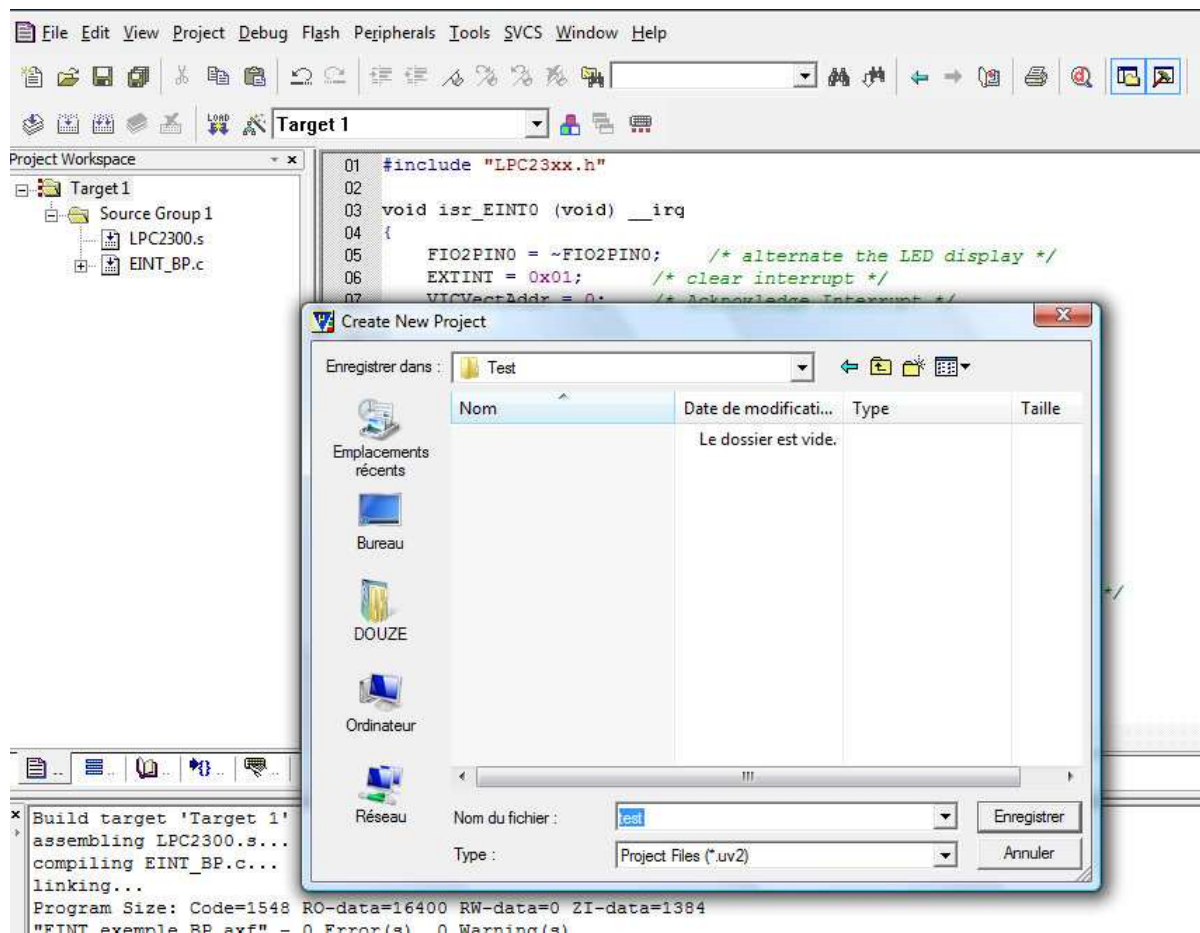
Tout d'abord, il faut ouvrir le logiciel  $\mu$ Vision3 :

Démarrer → Programmes → Keil  $\mu$ Vision → Keil  $\mu$ Vision3

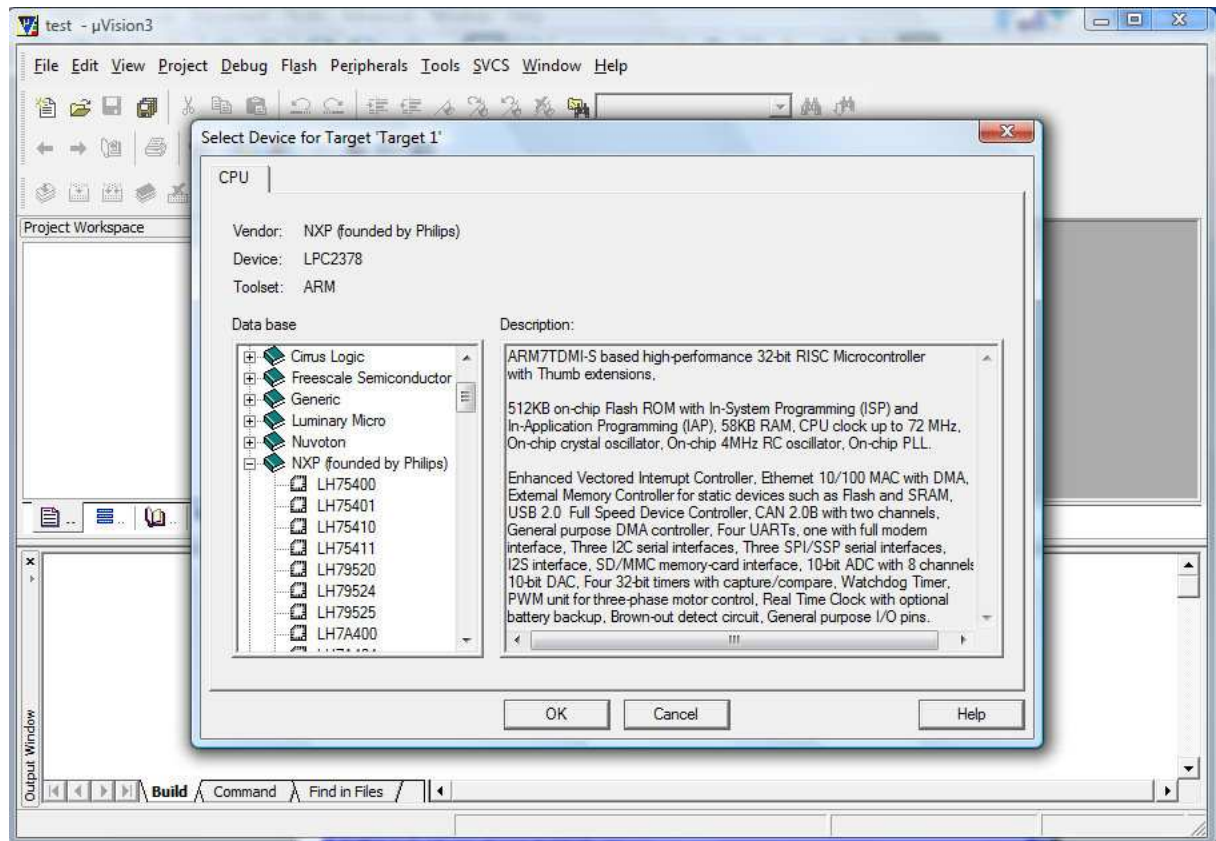
La réalisation d'un programme, sous cet environnement, nécessite la création d'un projet. Le menu Project de  $\mu$ Vision3 permet de créer, ouvrir et éditer des projets. Un projet contient l'ensemble des fichiers source (C ou assembleur et les bibliothèques logicielles) de l'application ainsi que des informations concernant les outils associés à chaque fichier. Il contient également les informations concernant le système cible (différentes adresses d'implantations de la table des interruptions, du programme et de la zone de données externe). Pour créer un nouveau projet allez dans :

Project → New Project

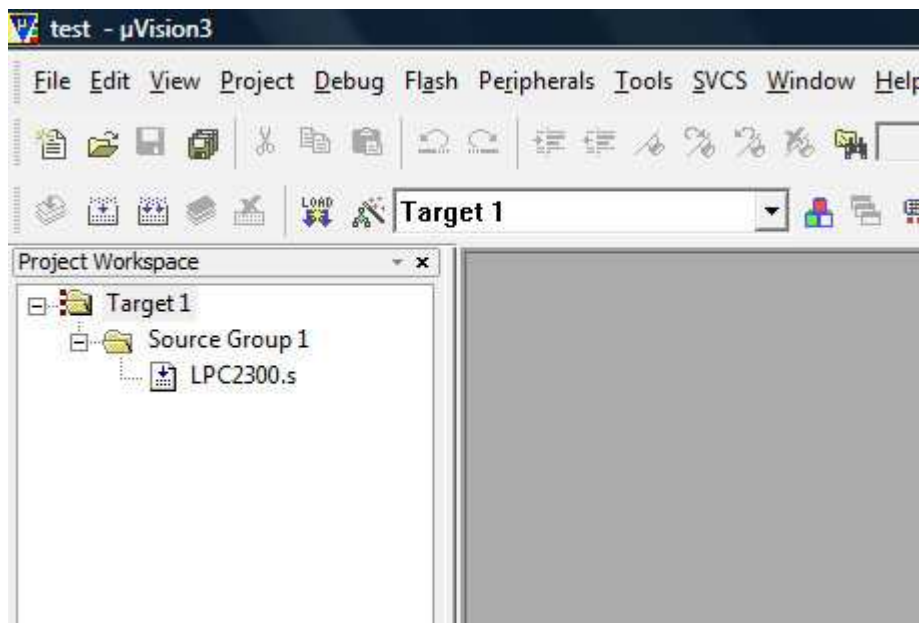
Choisissez le dossier dans lequel vous voulez créer le projet et choisissez un nom pour le projet. Cliquez sur Enregistrer.




Une fenêtre s'ouvre qui vous propose de choisir la cible. Pour les TPs, choisissez dans la gamme NXP le cible LPC2378. Cliquez sur OK.




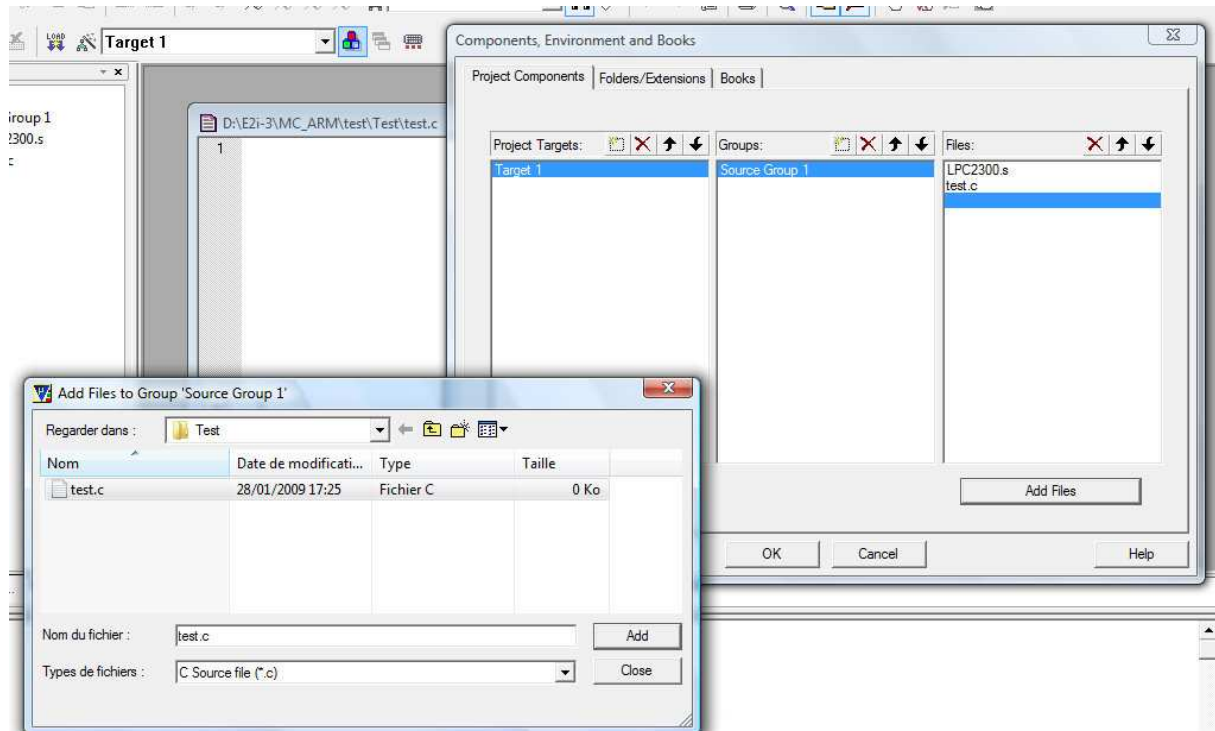
Ensuite une nouvelle fenêtre s'ouvre qui vous demande : « Copy Philips LPC2300 Startup code to project Folder and Add File to Project ? ». Répondez Oui. L'IDE rajoute le fichier LPC2300.s à votre projet. Ce fichier permet d'initialiser les registres internes et la mémoire de votre µC.



Une fois que vous avez créé un nouveau projet, vous pouvez éditer votre code ou ajouter des fichiers à votre projet .

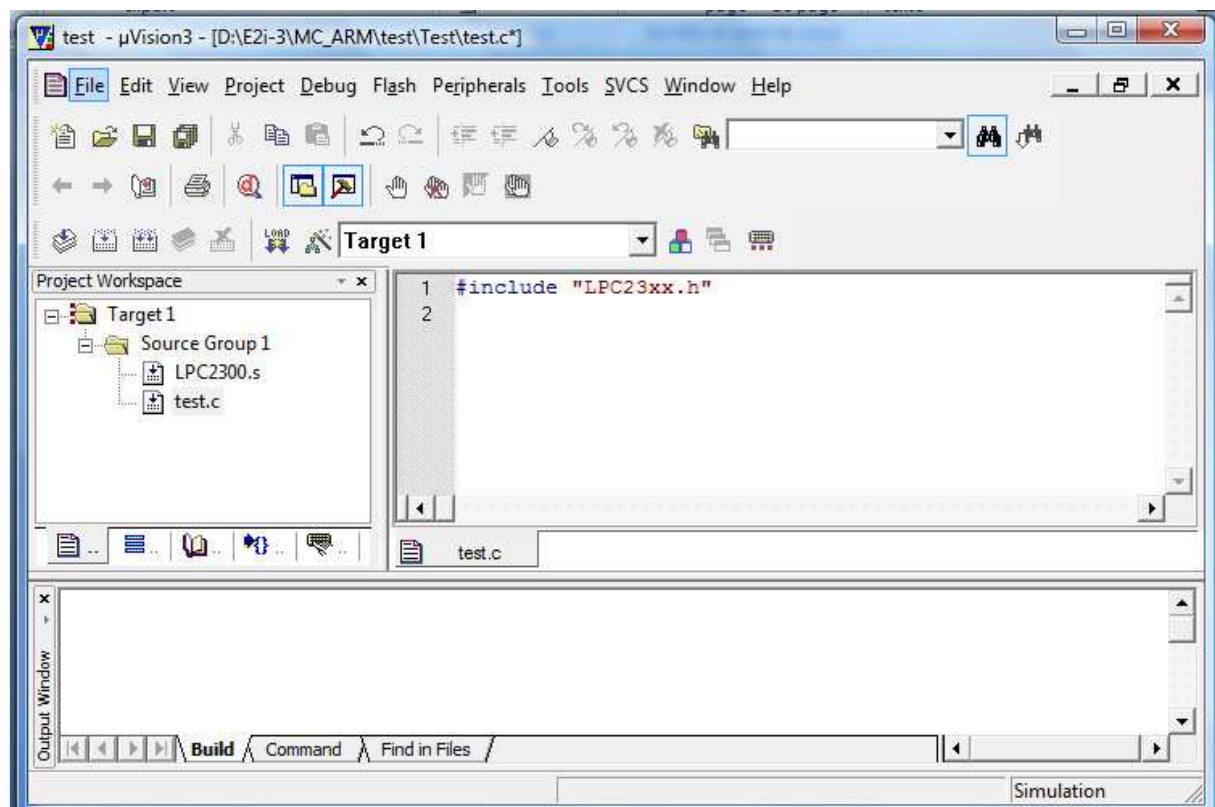
Pour créer un nouveau fichier, cliquez sur l'icône  (en haut à gauche) puis sauvegardez-le dans le même répertoire que le projet avec l'extension .c pour un fichier écrit en langage C et avec l'extension .s pour un fichier écrit en assembleur.

Ensuite, ajoutez le fichier ainsi créé à votre projet. Cliquez sur l'icône . Cliquez ensuite sur le bouton Add Files et sélectionnez le fichier .c à ajouter au projet. Cliquez sur Add.

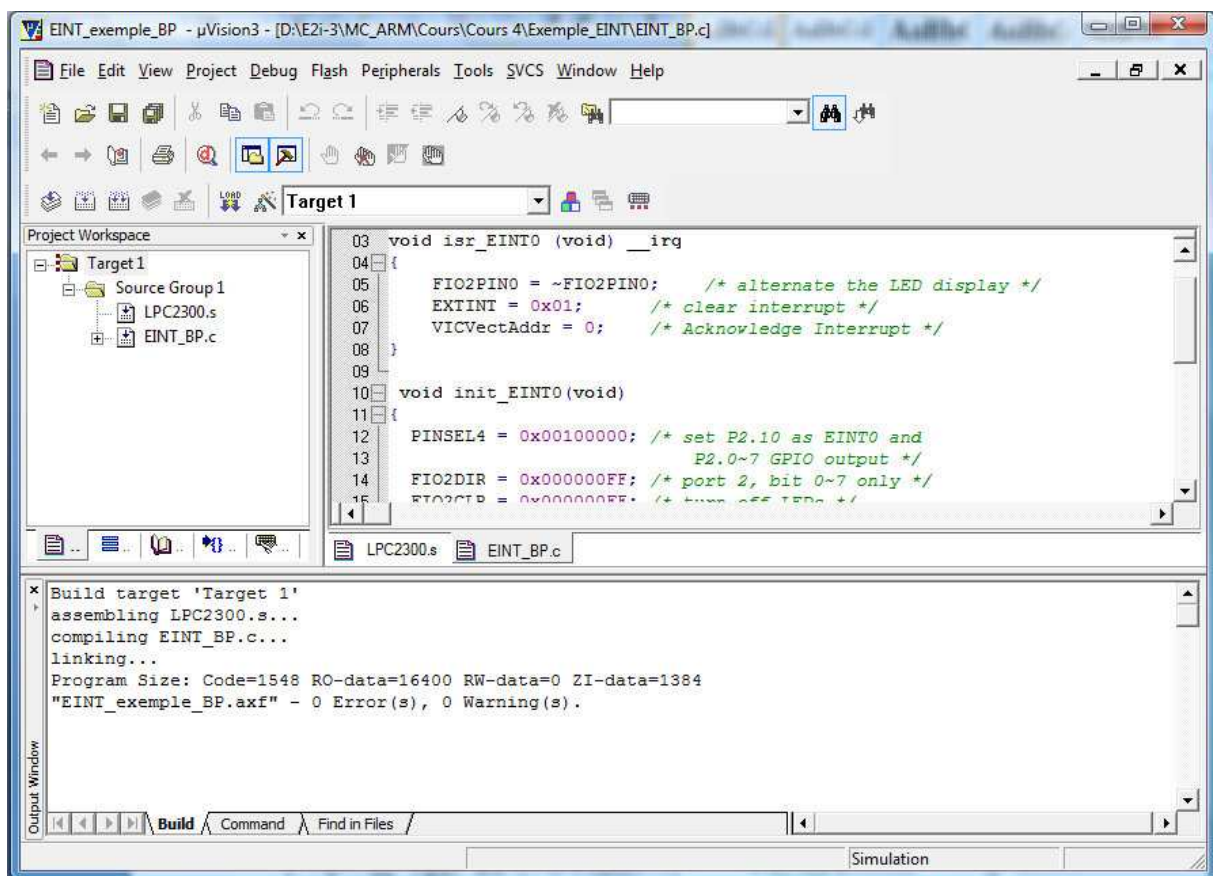
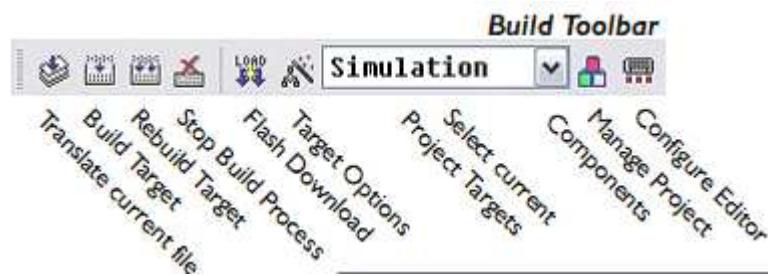


Maintenant, vous devriez avoir quelque chose qui ressemble à la copie d'écran comme ci-dessous et vous pouvez commencer à éditer votre code.


**Attention :** afin d'utiliser directement le nom des registres, le programme principal devra toujours inclure le fichier *LPC23xx.H* qui contient toutes les adresses des registres du microcontrôleur.

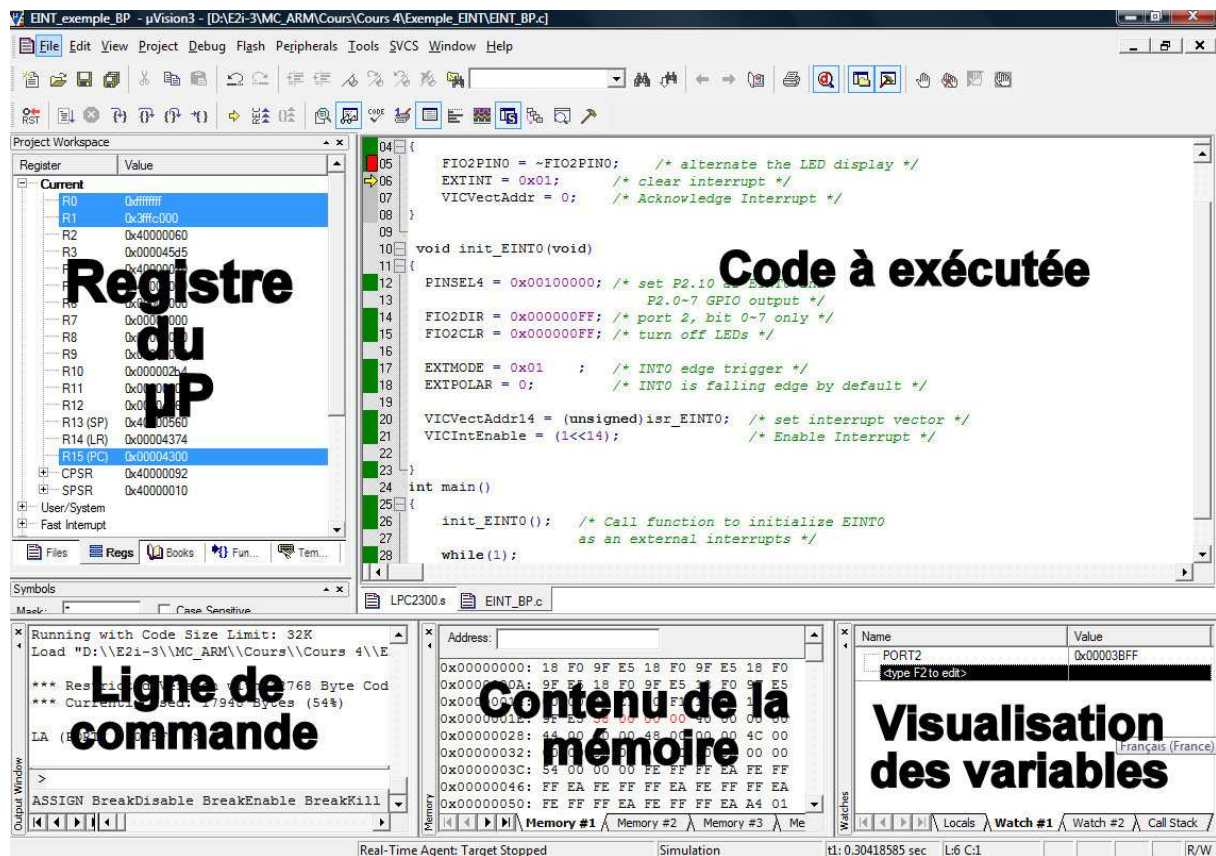








## 6- Simulation


Une fois que vous avez compilé votre code, vous pouvez le simuler en cliquant sur l'icône . Ceci lance l'émulateur de votre cible (LPC2378) qui permet de simuler votre application comme si elle était dans la cible.




La flèche jaune représente la prochaine instruction qui sera exécutée. Le carré rouge indique un point d'arrêt. Lorsque une instruction est précédé par un carré vert cela veut dire que cette instruction a déjà été exécuté au moins une fois (code coverage).

Pour exécuter une seule instruction, en mode pas à pas, cliquer sur l'icône : 

Un double click sur la zone grisée à la gauche d'une ligne de code permet d'insérer ou de retirer un point d'arrêt (breakpoint). Les points d'arrêts sont très utiles pour faire du déverminage (debug), vous placez des points d'arrêts aux endroits importants et ensuite vous pouvez vous déplacer d'un point d'arrêt à l'autre en cliquant sur le bouton run 

Pour exécuter le programme, en mode continu (run), cliquer sur l'icône : 

Pour suspendre l'exécution, cliquer sur l'icône : 

Vous pouvez suivre la valeur des différentes variables déclarer dans votre programme dans la fenêtre Watch que vous pouvez visualiser / cacher en cliquant sur l'icône : .

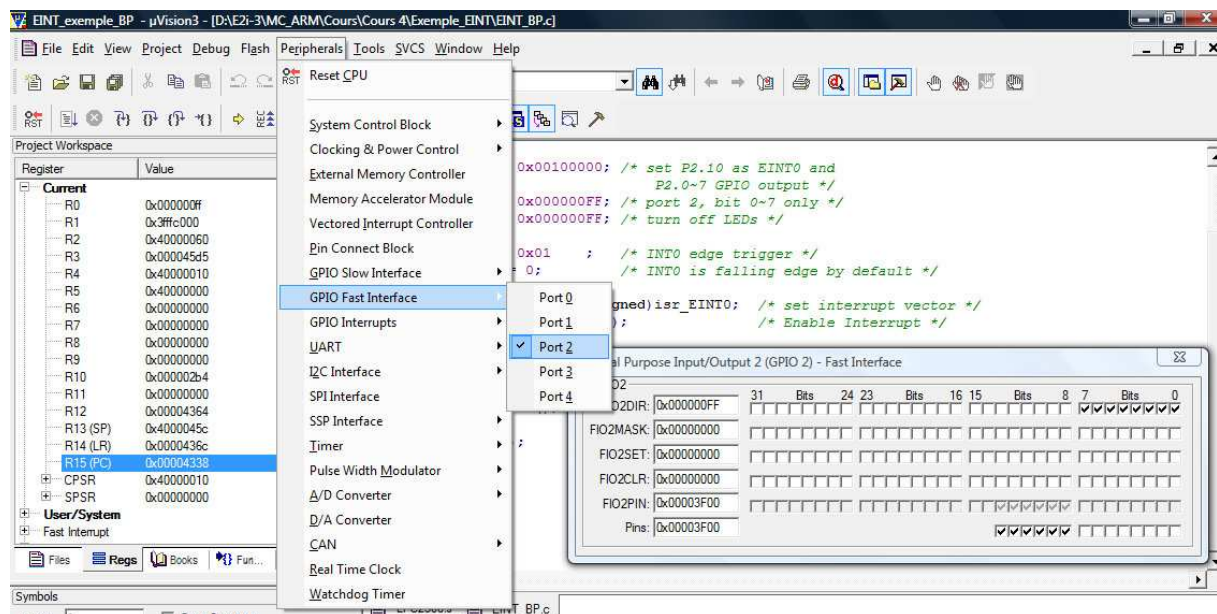
- Pour visualiser les variables locales aller dans l'onglet 'Locals'.
- Pour visualiser les variables globales aller dans l'onglet 'Watch #1', appuyez sur la touche F2 et entrez le nom de la variable à visualiser.


- Vous pouvez aussi visualiser la valeur des variables en positionnant le curseur dessus et attendre 1 ou 2 secondes.


Si vous voulez voir contenu de la mémoire interne, cliquer sur l'icône : 



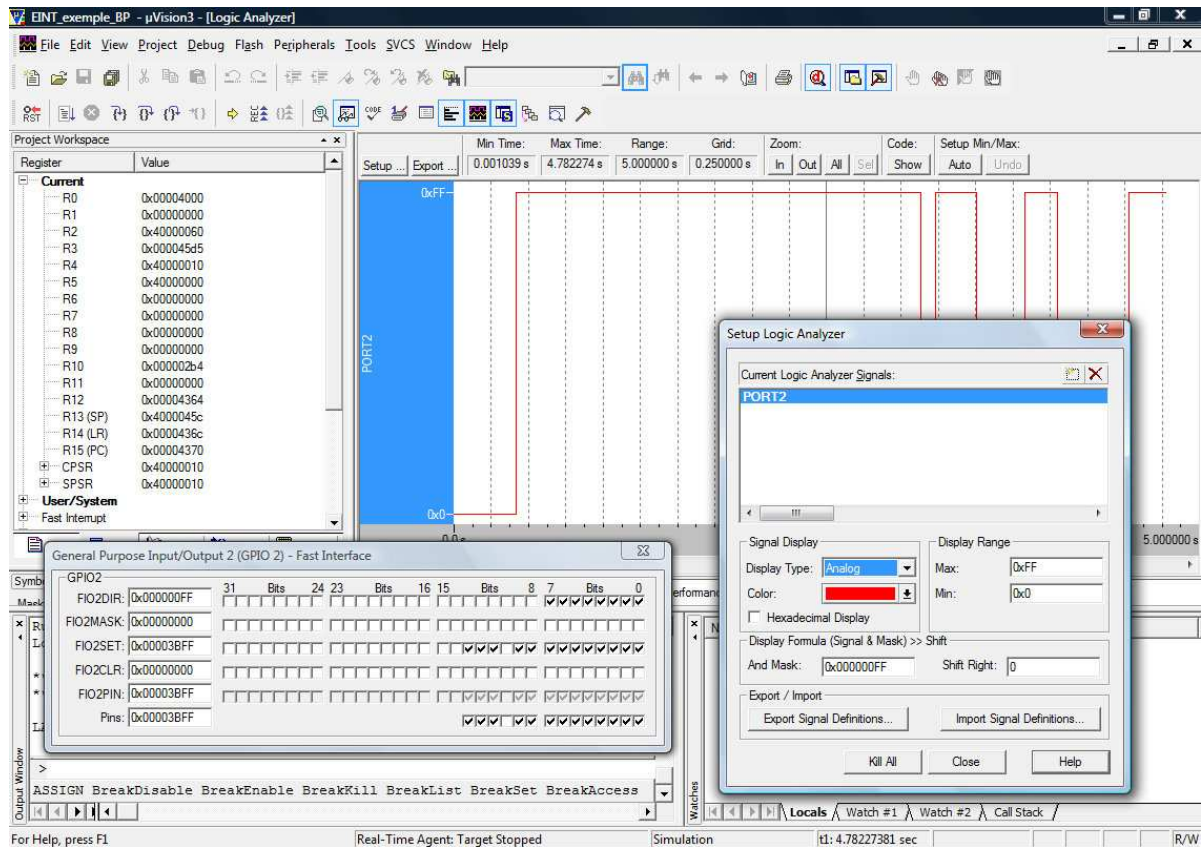
Le menu **Peripherals** vous permet de consulter (ou de modifier) l'état de tous les registres du microcontrôleur. Ci-dessous, la fenêtre correspondant au port parallèle GPIO2.





Il est également possible de tracer le chronogramme des variables du programme grâce à la fenêtre Logic Analyzer. Cliquez sur l'icône . Pour ajouter un signal dans cette fenêtre, il existe 2 méthodes :

- Soit en ligne de commande. Pour ajouter une variable à visualiser, écrire `LA nom_variable` (Exemple : `LA test`) dans la fenêtre de commande de Keil. Pour rajouter un signal, écrire `LA nom_signal`. Pour réinitialiser le chronogramme, taper la commande `LA kill *`.
- Soit en utilisant l'interface graphique. Cliquez sur **Setup** (en haut à gauche). Pour ajouter un signal, cliquez sur l'icône . Rentrez le nom de la variable ou du signal à visualiser. Par exemple `PORT2` pour le GPIO2. Vous avez la possibilité de faire un masque (champ « **And Mask** ») pour voir uniquement une partie du port ou encore vous pouvez limiter la valeur max (champ « **Max** »).





L'icône Serial Window  vous permet de visualiser les données émises ou reçues sur la liaison RS232.

L'icône Disassembly Window  permet de voir le code assembleur généré par le compilateur.